

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«СЕВЕРО-КАВКАЗСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт цифрового развития
Кафедра инфокоммуникаций

**ОТЧЕТ
ПО ЛАБОРАТОРНОЙ РАБОТЕ №2.5**

Выполнила:
Кузнецова Алена Валерьевна
1 курс, группа ИВТ-б-о-21-1,
09.03.01 «Информатика и
вычислительная техника»,
направленность (профиль)
«Программное обеспечение средств
вычислительной техники и
автоматизированных систем», очная
форма обучения

(подпись)

Отчет защищен с оценкой _____ Дата защиты _____

Ставрополь, 2022 г.

Цель работы: приобретение навыков по работе с кортежами при написании программ с

помощью языка программирования Python версии.

1. Создать общедоступный репозиторий на GitHub, в котором будет использована лицензия MIT и язык программирования Python.

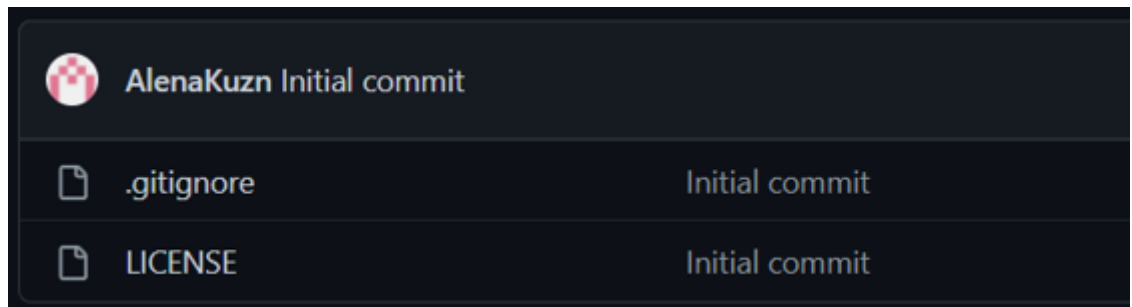


Рисунок 1 – Созданный репозиторий

2. Выполнили клонирование созданного репозитория.

```
C:\Users\akuzn>cd /d C:\Users\akuzn

C:\Users\akuzn>git clone https://github.com/AlenaKuzn/2.5.git
Cloning into '2.5'...
remote: Enumerating objects: 4, done.
remote: Counting objects: 100% (4/4), done.
remote: Compressing objects: 100% (4/4), done.
remote: Total 4 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (4/4), done.
```

Рисунок 2 – Клонирование репозитория

3. Организовали свой репозиторий в соответствии с моделью ветвления git-flow.

```
C:\Users\akuzn\2.4\individ>git flow init

Which branch should be used for bringing forth production releases?
- main
Branch name for production releases: [main]
Branch name for "next release" development: [develop]

How to name your supporting branch prefixes?
Feature branches? [feature/]
Bugfix branches? [bugfix/]
Release branches? [release/]
Hotfix branches? [hotfix/]
Support branches? [support/]
Version tag prefix? []
Hooks and filters directory? [C:/Users/akuzn/2.4/.git/hooks]
```

Рисунок 3 – Организация по модели ветвления git-flow

Контрольные вопросы:

1. Что такое кортежи в языке Python?

Кортеж – это неизменяемая структура данных, которая по своему подобию очень похожа на список.

2. Каково назначение кортежей в языке Python?

Существует несколько причин, по которым стоит использовать кортежи вместо списков. Одна из них – это обезопасить данные от случайного изменения. Если мы получили откуда-то массив данных, и у нас есть желание поработать с ним, но при этом непосредственно менять данные мы не собираемся, тогда, это как раз тот случай, когда кортежи придутся как нельзя кстати.

Используя их в данной задаче, мы дополнительно получаем сразу несколько бонусов – во-первых, это экономия места. Дело в том, что кортежи в памяти занимают меньший объем по сравнению со списками.

3. Как осуществляется создание кортежей?

Кортеж с заданным содержанием создается также как список, только вместо квадратных скобок используются круглые. При желании можно воспользоваться функцией `tuple()`.

4. Как осуществляется доступ к элементам кортежа?

Доступ к элементам кортежа осуществляется также как к элементам списка – через указание индекса. Но, как уже было сказано – изменять элементы кортежа нельзя!

5. Зачем нужна распаковка (деструктуризация) кортежа?

Обращение по индексу, это не самый удобный способ работы с кортежами. Дело в том, что кортежи часто содержат значения разных типов, и помнить, по какому индексу что лежит — очень непросто.

6. Какую роль играют кортежи в множественном присваивании?

7. Как выбрать элементы кортежа с помощью среза?

8. Как выполняется конкатенация и повторение кортежей?

9. Как выполняется обход элементов кортежа?

10. Как проверить принадлежность элемента кортежу?
11. Какие методы работы с кортежами Вам известны?
12. Допустимо ли использование функций агрегации таких как `len()` ,
`sum()` и т. д. при работе
с кортежами?
13. Как создать кортеж с помощью спискового включения.